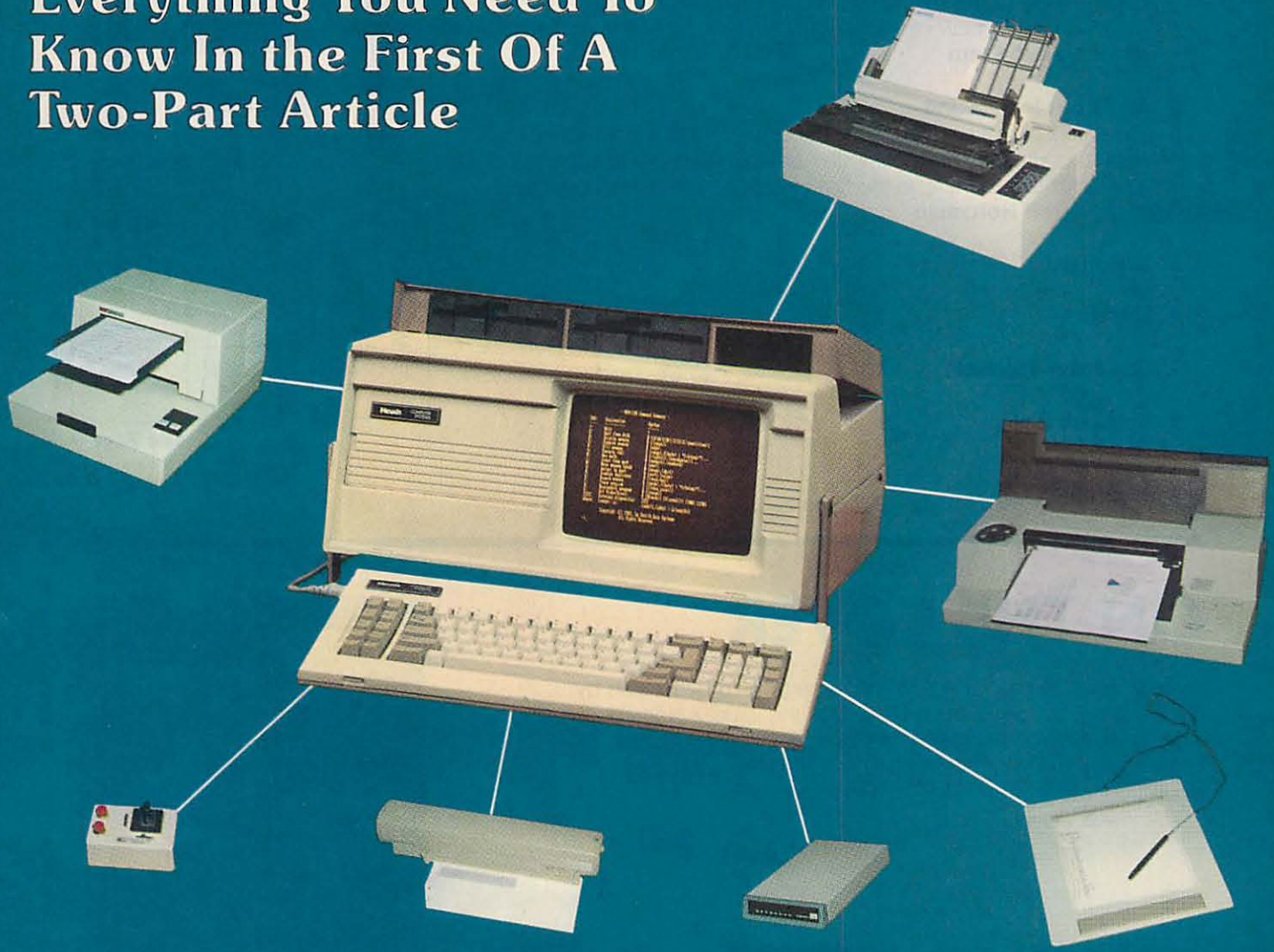
VOL. 3 NO. 12 DEC. 1986

COMPUTERDIGEST

A NEW KIND OF MAGAZINE FOR ELECTRONICS PROFESSIONALS

ALL ABOUT INTERFACING

Everything You Need To Know In the First Of A Two-Part Article



V-20 VS 8088

A New Battle Is Shaping Up

A GERNSBACK PUBLICATION

CONTENTS

Vol. 3 No. 12 December 1986

6 V-20 vs 8088
There's a new battle shaping up on the computer horizon. And when something new happens, we want to make sure you know about it. **Marc Stern**

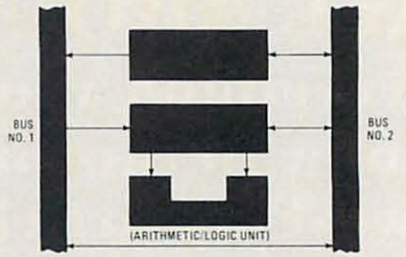
9 All About Interfacing
This article has been a long time in coming. No matter what you want to connect to your computer, or what you want to connect your computer to, here, in the first of a 2-part article is all you need to know. **Jeff Holtzman**

3 Editorial

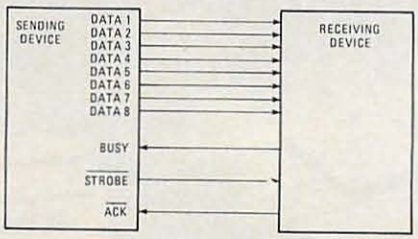
4 Letters

4 Computer Products

5 Software Review



See Page 6



See Page 9

ON THE COVER

If you're going to be doing ANY interfacing, you'd better start by reading the story beginning on Page 9 of this issue. Admittedly, it's a bit more involved than laying out photos and drawing the connecting lines! Most of the photos of assorted equipment around our office, were taken by Herb Friedman.

COMPUTERDIGEST
A NEW KIND OF MAGAZINE FOR ELECTRONICS PROFESSIONALS
ALL ABOUT INTERFACING
Everything You Need To Know In the First Of A Two-Part Article
V-20 VS 8088
A New Battle Is Shaping Up

COMING NEXT MONTH

Naturally, we try to pack all the information that we can into each issue. But in the January, 1987 Issue, we've really outdone ourselves. You won't want to miss the conclusion of our two-part article on interfacing, and we've got an important how-to on TVRO antenna pointing. We're finishing up with a fine piece on a five-volt RS-232. Don't miss it.

EDITORIAL

Let's standardize the "standards."

■Henry Ford has to be the father of mass production, and he built this scheme on totally-interchangeable parts. The carburetor of one car could easily be switched with the carburetor of another car. And the need for standards extends well-into electronics. Every young electronics devotee knows the standard for the resistor color code. You can see them counting on their fingers, as they recite, "Bad boys..."

But something began to go awry when it came to stadardizing capacitor color codes. There were decimals to be considered, and multipliers, and the result was utter confusion.

Today, in the computer field, we have the so-called "RS-232" standard. But the manufacturers do not cleave to the standard, using their own variations for their own convenience. The result is a non-standard standard, and when you say "We're using RS-232," you find people asking just *which* "RS-232?"

It goes even beyond that. We have assorted operating systems, beginning with MS-DOS and CP/M and a host of other proprietary systems produced by various manufacturers, that prevent a software package that works on one, from working with another. The standard that seems to be emerging, is the IBM system, for almost every software manufacturer talks about "IBM compatibility."

Even the simple blank disk cries for standardization. You now have a choice of "hard sectored disks," or "soft sectored disks."

Wouldn't it be nice to be able to buy a computer without having to trace down the standards to make sure that they were indeed standard? Or that your projected computer purchase would operate *all* the disks you now have? Sure it would.

Now I'm shopping for a videocassette recorder. Top loading? Front loading? VHS? Beta? Programmable? Fast scan?

Decisions! All day long, decisions!

Byron G. Wels
Editor

ComputerDigest is published monthly as an insert in Radio-Electronics magazine by Gernsback Publications, Inc., 500-B Bi-County Blvd., Farmingdale, N.Y. 11735. Second-Class Postage Paid at New York, N.Y. and additional mailing offices. Copyright © 1986 Gernsback Publications, Inc. All rights reserved. Printed in U.S.A.

A stamped self-addressed envelope must accompany all submitted manuscripts and/or artwork or photographs if their return is desired should they be rejected. We disclaim any responsibility for the loss or damage of manuscripts and/or artwork or photographs while in our possession or otherwise.

COMPUTER DIGEST

M. Harvey Gernsback,
editor-in-chief, emeritus

Larry Steckler,
EHF, CET: publisher & editor in chief

Art Kleiman,
editorial director
Byron G. Wels,
editor
Brian C. Fenton,
managing editor
Carl Laron,
associate editor
Robert A. Young,
assistant editor
Jeff Holtzman
technical editor
Teri Scaduto Wilson
editorial assistant
Ruby M. Yee,
production director
Karen Tucker,
production advertising
Robert A. W. Lowndes,
production associate
Geoffrey S. Weil,
production assistant
Andre Duzant,
technical illustrator
Jacqueline P. Cheeseboro
circulation director
Arline R. Fishman,
advertising director

Computer Digest
Gernsback Publications, Inc.
Executive offices
500-B Bi-County Blvd.,
Farmingdale, NY 11735
516-293-3000
President: Larry Steckler
Vice President: Cathy Steckler

ADVERTISING SALES 516-293-3000
Larry Steckler
Publisher

NATIONAL SALES
Joe Shere
1507 Bonnie Doone Terrace
Corona Del Mar, CA 92625
714-760-8967

LETTERS

Too Long

I submitted an article to your magazine, and got paid for it. Thank you. That was several months ago, and it still hasn't appeared in print. What's the delay?—S.T., Orlando, FL.

Have a heart! First of all, things just take time. For example, this column will actually appear in the December Issue, but as I write these words, it's really August. The 17th, to be exact. And yes, if you look it up on a calander, it's a Sunday. And right now, the magazine is all set through February. So maybe the next time we need a three-pager, and yours fits, you might get lucky.

Warranty

I'm confused about my computer warranty. It runs for a full year, *unless* I open the case! Now (a) how can I add a board or do some construction if I don't open the case, and (b) how are they going to know whether or not I did?—J.T., Fresno, CA.

You pays yer money and takes yer cherce. Open it up, and you automatically void the warranty. And yes, they will know, for the manufacturers usually put a dab of paint on a cover-mounting screw, or paste on a small paper patch. Break the paint or paper and you kiss your warranty goodbye. We usually suggest waiting until the warranty expires before experimenting.

Budding Editor?

I'm fascinated with the publishing business and would (I think) like to get into it when I get older. What would you recommend?—A. L., Memphis, TN.

It's been extremely kind to me, and I'd suggest you start preparing the publishing business as well as yourself, by submitting articles on a freelance basis. This will help you learn to write, and will start getting your name around.

Expensive?

I've been pricing furniture (desks/tables) for my computer equipment. Now why is that stuff so expensive?—S.K., Reno, NV.

It doesn't have to be. Your local lumber yard can sell you a sheet of 4 × 8 plywood, finished one side and ¾-inch thick at a very low price. For another couple of bucks they'll cut 30-inches off one end and run another cut 30-inches down the length of the remainder. That will give you two handsome table tops, ready-to-finish and enough wood to make legs with!

COMPUTER PRODUCTS

For more details use the free information card inside the back cover

VOICE-RECOGNITION SYSTEM, the *Voice Master,* is a half-card expansion board with resident program that recognizes hundreds of words. It is designed for the IBM-PC, XT, AT and compatible

CIRCLE 27 ON FREE INFORMATION CARD

personal computers. Versions are currently available for Commodore, Apple II series, and Atari 8-bit microcomputers.

The *Voice Master* has a suggested retail price of $129.95, which includes half-card circuit board, microphone headset, and resident system software.—**Covox, Inc.,** 675-D Conger St., Eugene, OR 97402.

SYNCHRONOUS MODEM

The model *91* is designed to function point-to-point or in a multi-polled network. Multi-dropped operation is achieved by using a daisy-chained approach to system wiring, in which each model *91* regenerates each signal before passing it on in the link. That process of incorporating signal repeaters results in extremely-high data reliability, especially at long distances. At 19,200 baud, the model *91* will support link distances of one mile.

The opto-isolation feature assures that devices connected to the model *91* are electrically isolated from each other, thereby eliminating problems associated with potential differences between grounds.

The model *91* provides full duplex synchronous serial data communication from 150 to 19,200 baud using frequency modulation. Transmission of a self-clocking FM signal on a balanced

CIRCLE 28 ON FREE INFORMATION CARD

opto-isolated current-loop allows transfer of clocking information as well as data. The model *91* can also be linked to establish a full-duplex bus network between host and a number of remote terminals. It is priced at $250.00.—**Telebyte Technology, Inc.,** 270 Pulaski Road, Greenlawn, NY 11740.

SOFTWARE REVIEW

That's right! A print shop in your computer.

■Judging by its use, The Print Shop is probably the best-selling program. The program is used by schools, supermarkets, shops, even used-car dealers (and you know they're fussy about advertising).

For all its popularity, The Print Shop isn't a word processor, a spread sheet, time-manager or CAD/CAM. It's a signmaker. Using pin-feed paper it makes 8½ × 11 inch signs, greeting cards from folded 8-1/2 × 11 inch paper, letterheads, and banners; which can be used as advertising flyers, bookplates, report covers, award certificates, even abstract printed designs.

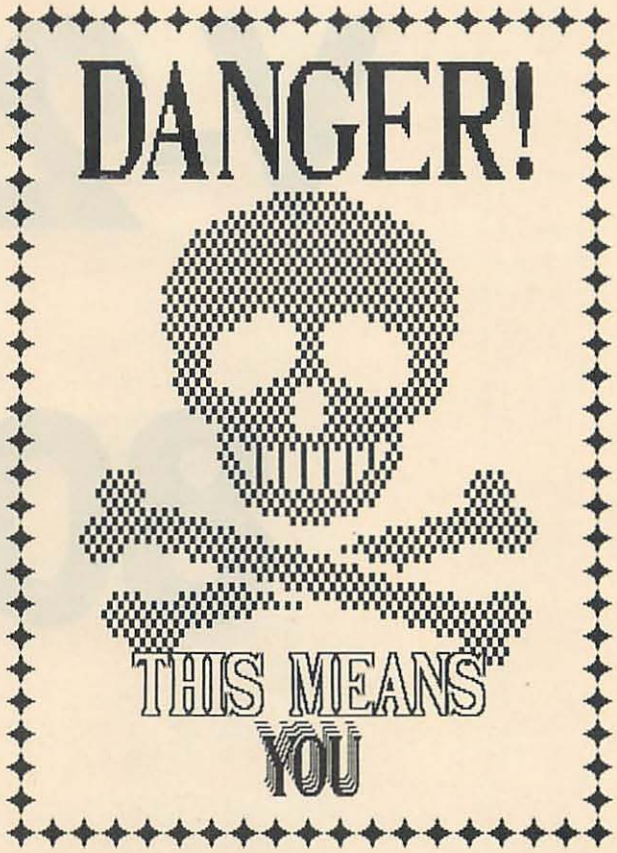
The program has an assortment of type styles in three formats (solid, outline and three-dimensional), borders, graphic characters and symbols, even abstract patterns. Each type style is its own size, each can be stretched to a proportional GIANT size. (The Commodore version has eight upper case fonts, nine border styles, and 60 graphics. The IBM version has twelve type styles in upper and lower case, sixteen borders, and 119 graphics.)

The user can integrate any type style with any border and any graphic, or any print function can be left out so the paper can be run through for a second or third printing, or run through to add conventional text from a word processor. When set to print greeting cards, all type, graphics and borders are reduced to the appropriate size and the printing is automatically rotated 180° so that everything comes out right side up when the paper is folded into a card.

Because the program was intended for children (if adults ever give them a chance to use it), it is menu-driven by cursor movement: the user steps the cursor to the desired mode and then hits RETURN. Each menu, except for graphics selection, shows the available and selected border and type styles. (The IBM version also shows the available and selected graphics.)

Special screens let the user select a small, medium or large graphic, built-in or custom graphic layouts, and line and page type-setting. The screen shows the possible graphic locations depending on the user's size selection (the graphics cannot overlap), and the allowable characters per line for each type and size type. If you select GIANT size and the characters won't fit the available space the screen shows what will fit, along with left, center or right positioning normal, reverse or 3-D type on a per line basis, and automatic top-to-bottom centering, if desired.

The program is goof-proof, and you can step back through each function one at a time and change only



what you want to change without losing all your work. For example, if after completing a sign you decide to change the border, say from *stars* to *hearts*, you step back through the composing, type selection, and graphics until you come to the border menu. You then change the border and step forward through graphics and type without affecting anything other than the border selection. You can even step back and change the position of a single line of type, or its outline, or even erase a line or characters (there is a built-in editor). What you can't do is mix type styles or graphics. The only variation on this rule is the IBM version, which permits the type to be upper or lower case on a per-line basis; that is, the line must be all upper case or all lower case.

The Print Shop has a graphics editor that lets you create your own graphics. In the Commodore version, the user-generated graphics can be saved to disk, but that's all that can be saved. In the IBM version the graphics and any sign, card, stationary or banner design can be saved to disk.

Each version supports a selection of printers, which is listed on the back of the box. If your printer isn't listed assume the program will not work.

The Print Shop is supplied on disk and requires at least one disk drive. It comes with a supply of heavyweight pinfeed paper and greeting card envelopes. Additional multi-color supplies and disk volumes of additional graphics characters are available. The Print Shop is heavily discounted: It sells for $19.95 to $59.95 depending on the version and the dealer.—Broderbund Software, 17 Paul Drive, San Rafael, CA. 94903-2101.◀●▶

V-20 vs 8088

Is there a direction change in the offing?

MARC STERN

■The way to stay afloat in the microcomputer world is IBM emulation, as it is accepted by big business, professionals, and home users.

Many manufacturers, having first tried to go their own ways found that crowds weren't flocking to their doors for solutions. The crowds were still heading toward IBM. Those firms that were able to switch have survived, while those that haven't are no longer active in the microcomputer market.

The ultimate in emulation would be a microprocessor that not only emulates, but surpasses that used in the IBM Personal Computer series.

Such a microprocessor is the Nippon Electric Co.'s (NEC) V20/30 series. It not only emulates the 40 pin Intel 8086/8088 series, used in the IBM PC and close compatibles, but it betters that series by using less power (300 mW versus 1.7 W) and operates about 20 percent faster than the IBM version.

The only way you can differentiate between the two, is the NEC stamp on the microprocessor chip, and the "V" designation.

The V20/30 series has an added mode, 8080 emulation. This gives CP/M users an out because CP/M was written to work in a Z80/8080 environment. With an emulator the V-20 will function effectively as an 8080. So those CP/M programs which may have been threatened with obsolesence by the 8086/8088 series and PC/MS-DOS will gain a new life and users will be able to retain their investment in software.

A closer look

The V20 and V30 are equivalent to the 8088 and 8086, respectively. Like the 8088, the V20 is an 8/16 microprocessor. It has an 8-bit data bus, but a 16-bit internal architecture. The marketing departments of various microcomputer manufacturers like to call this type of microprocessor a 16-bit device, but, it really isn't.

The true 16-bit device is the 8086. It has a 16-bit architecture, and also a 16-bit data bus. It is faster than the 8088 in realtime number crunching. The V30 is the

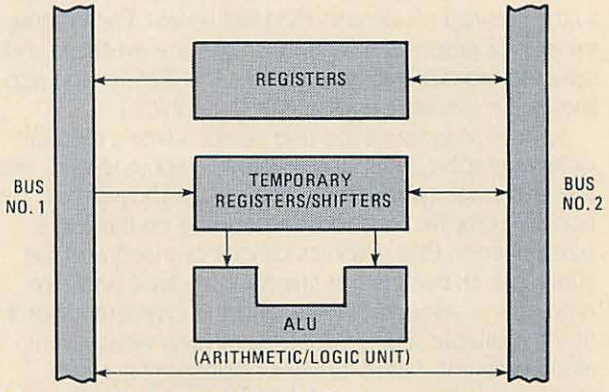


Fig. 1: A DUAL INTERNAL BUS helps increase the performance level of the V20/V30 series. The 8086/8088 family has a single internal bus and must suspend information processing as it moves source and destination data back and forth. The V20/V30 can move this data simultaneously.

functional equivalent to this device.

Both microprocessors are available in 5 and 8 MHz versions and they are CMOS devices. While both microprocessor series are functionally equivalent, the V20/30 series uses 600 percent less power to handle the same jobs. For a system that is struggling at the razor edge of its power supply the lower demands of the NEC device can make a difference.

Not only are the V20/30 chips power misers in their active mode, they offer a standby mode where their power consumption drops from 300 mW to 30 mW which means these devices can be used in standard desktop machines, and they can be used in briefcase or lap-top microcomputers.

The advantages seem to be with the V20/30, because the V20/30 series is much newer than the 8086/8088 family. Because it was designed recently, NEC has taken advantage of Very Large-Scale Integration and gate array technology, as well as advances in CMOS technology to produce this capable chip. The 8086/8088 series was designed in the mid-1970s and its age is beginning to show.

Another result of the age difference is the V20/30's dual internal bus (See Fig. 1), which compares with the single internal bus of the 8086/8088 family. This can effectively increase the speed specifications of the V20/V30 because destination and source data can move along the data bus at the same time in two directions.

Data are taken into the registers in preparation for processing and the data are sent to the temporary storage registers and shifters. At this point, the data goes to the arithmetic-logic unit (ALU) for processing and then to either bus, depending on whether the information is source or destination. Data can pass from the registers back to the second bus.

Moving in parallel, the microprocessor is freed of of wait states generated by having to free alternate cycles on the data bus for 16- bit source data and then destination data. Picture a single freeway lane trying to accommodate traffic in both directions during rush hour. This applies to the execution time of the 80886/8088 family. (We're talking about a difference of microseconds and the user won't notice the difference in speedup unless the application crunches a lot of numbers and must constantly fetch and latch the numeric data.)

Another change in the architecture of the V20/V30 is the substitution of hardware address calculations for the software (microcode) and hardware address calculation of the 8086/8088 series. The V20/30 series includes a hardware section which performs address calculations from 2.6 to 6 times faster than the 8086/8088. Where it takes the 8086/8088 from five to 12 clock cycles to perform address calculations, the V20/V30 performs them in two. Address calculation, then, is also transparent to applications, which means that a developer only has to be concerned with a standard set of addresses for a particular device, rather than having to worry about both hardware and software addressing.

Another performance enhancement that is unique to the V20/V30 is its prefetch pointer. (See Fig. 2)

Working with the instruction pointer, the prefetch pointer speeds things by moving to the next instruction and points to it as the microprocessor works through a program. A movable pointer, the prefetch jumps to the next instruction no matter how far it is from the instruction pointer. This enhances performance because the next instruction is always available for the micro.

The V20/V30 begins with performance advantages that the 8086/8088 series doesn't have. These advantages wouldn't make much difference if the V20/V30 wasn't compatible with the 8086/8088 family, which it is.

When it was developed, this series was modeled on and implemented the instruction set of the 80186/80188, a more powerful and later version of the 8086/8088 family. The differences were in the level of integration of such functions as the timer and interrupt controller, which are on-chip functions on the 80186/80188 series. The instruction set is enhanced with new instructions which add power and flexibility, but without sacrificing compatibility. There's no loss of compatibility as the complete instruction set of the 8086/8088 series is supported, as well as the additional code.

Speed improvement

While the specifications indicate a potential speed increase, it doesn't work out that way in practice. The V20/V30 series gives a user about a 20 percent increase in performance. Independent testing has confirmed this. The increase in speed is noticeable in processor-intensive applications where the microprocessor must constantly issue fetch commands for new instructions. The same testing has confirmed that input-output intensive applications, such as word-processing or telecommunications won't benefit that much from the new microprocessor because the system is constantly waiting for keyboard or communications port input.

You might be wondering why the speed differential exists. The microprocessor is busy doing other things—adding, subtracting, looping—while it is handling specific data addressing and calculation. Because it is, its resources are being spread through the system and they can't all be brought to bear for the ultimate speed increase.

Clock speed also seems to enter the picture. Running

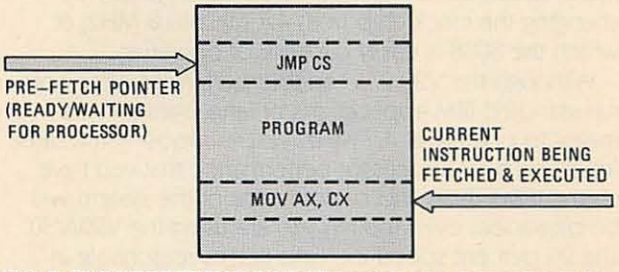
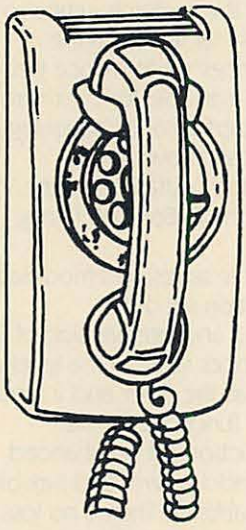


FIG. 2: THE PREFETCH POINTER speeds up processing in the V20/V30 series by moving directly to the next instruction to be executed by the microcprocessor. This speeds things up because the micro always has the next instruction ready for its instruction pointer, which doesn't have to move around as much.

R-E Computer Admart

Rates: Ads are 2¼" × 2⅛". **One** insertion **$825. Six** insertions **$800** each. **Twelve** insertions **$775.** each. Closing date same as regular rate card. Send order with remittance to **Computer Admart,** Radio Electronics Magazine, 500-B Bi-County Blvd., Farmingdale, NY 11735. Direct telephone inquiries to Arline Fishman, area code-**516-293-3000.** ***Only 100% Computer ads are accepted for this Admart.***



CALL NOW AND RESERVE YOUR SPACE

- 6 × rate $800.00 per each insertion.
- Reaches 239,312 readers.
- Fast reader service cycle.
- Short lead time for the placement of ads.

Call 516-293-3000 to reserve space. Ask for Arline Fishman. Limited number of pages available. Mail materials to: Computer Admart, RADIO-ELECTRONICS, 500-B Bi-County Blvd., Farmingdale, NY 11735.

ICs PROMPT DELIVERY!!!

SAME DAY SHIPPING (USUALLY)
QUANTITY ONE PRICES SHOWN for SEPT. 28, 1986

OUTSIDE OKLAHOMA: NO SALES TAX

640 Kbyte MOTHERBOARD KITS: Zenith 150, IBM PC/XT, Compaq Portable & Plus; hp Vectra

DYNAMIC RAM				
1Mbit	1000Kx1	100 ns	$40.00	$10.00
4464	64Kx4	150 ns	4.20	$12.00
41256	256Kx1	100 ns	4.85	8Mhz
41256	256Kx1	120 ns	3.40	8Mhz
41256	256Kx1	150 ns	2.75	V20
41128	128Kx1	150 ns	4.99	V30
4164	64Kx1	150 ns	1.35	
EPROM				
27512	64Kx8	250 ns	$22.00	$132.00
27C256	32Kx8	250 ns	6.50	$345.00
27256	32Kx8	250 ns	5.25	
27128	16Kx8	250 ns	3.95	5 Mhz
27C64	8Kx8	200 ns	4.75	8Mhz
2764	8Kx8	250 ns	3.95	
STATIC RAM				
43256L-12	32Kx8	120 ns	$27.00	8087
6264LP-15	8Kx8	150 ns	3.10	80287-8

OPEN 6½ DAYS. 7 AM-9 PM SHIP VIA FED-EX ON SAT.
SUNDAYS & HOLIDAYS: SHIPMENT OR DELIVERY, VIA U.S. EXPRESS MAIL.

SAT DELIVERY INCLUDED ON FED-EX ORDERS RECEIVED BY: Th: Std Air $6/4 lbs Fr: P-One $13/2 lbs

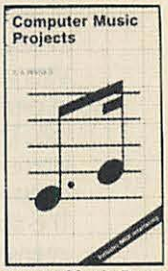
MasterCard VISA or UPS CASH COD
Factory New, Prime Parts µP∞
MICROPROCESSORS UNLIMITED, INC.
24,000 S. Peoria Ave., BEGGS, OK. 74421
(918) 267-4961

Please call for current prices because prices are subject to change. Shipping & insurance extra. Cash discount prices shown. Orders received by 9 PM CST can usually be delivered to you the next morning, via Federal Express Standard Air @ $6.00, or Priority One @ $13.00!

CIRCLE 61 ON FREE INFORMATION CARD

GETTING THE MOST FROM YOUR PRINTER

BP181—It is probable that 80% of dot-matrix printer users only ever use 20% of the features offered by their printers. This book will help you unlock the special features and capabilities that you probably don't even know exist. To order your copy **send $6.95 plus $1.50 for shipping** in the U.S. to **Electronic Technology Today Inc., P.O. Box 240, Massapequa Park, NY 11762-0240.**

COMPUTER MUSIC PROJECTS

BP173—Computer Music Projects shows how to use your home computer toproduce electronic music. Many circuits. Mostly jargon free. Send **$6.95 plus $1.75 shipping to ELECTRONIC TECHNOLOGY TODAY INC., PO Box 240, Massapequa Park, New York 11762-0240.**

A Z-80 WORKSHOP MANUAL

BP112—Starting with a review of computer principles, this book describes typical machine-code instructions followed by a detailed description of the Z-80 instruction set. Assembly language programming is also discussed with examples. Z-80 hex machine-code and assembler instructions are given in tabular form, along with in-our connections for the Z-80 and te associated devices....Order your copy from **Electronic Technology Today Inc., PO Box 240, Massapequa Park, NY 11762. Price is $6.95 plus $1.00 for shipping.**

a standard IBM Personal Computer or close compatible at its standard 4.7 MHz imposes timing and performance constraints on its 8088 processor. Reports circulated in the IBM user community for years of hyper-speed functionality brought about simply by changing the clock chip from 4.7 MHz to 8 MHz, at which the 8088 is easily capable of operating.

Although the V20 is a capable performer, when you run standard IBM applications or languages that are meant to operate at 4.7 MHz, you're imposing the same fetters on microprocessor performance that you have on the IBM's 8088 and performance of the system will be degraded, even though you are using the V20/V30. The 20 percent speed increase seems reasonable in light of this input.

Some users have complained about the speed shortfall and have questioned the capability of the V20/V30. But, those users have come to expect too much. It's one of the oldest marketing ploys in the microcomputer world, the specifications game, where the specifications are fantastic, but the performance doesn't match.

The solution, if you're considering the V20/V30, is knowledge. If you realize that the speeds won't be superfast, but will be modestly improved, then you'll get what you expect from the $25 to $30 V20/V30 chip.

Even a 20 percent performance increase is welcomed in some applications.

Finally, temperature and power usage are also important considerations and this is an area where the V20/V30 shines. It generates a lot less heat and uses manifestly less power than the 8088 and because it does it proves a blessing for a system overtaxed by the number of add-in cards and devices that may be on the motherboard because there's less heat and power consumption for the system box to contend with.◀◍▶

ALL ABOUT INTERFACING PART I

All you need to know about microcomputer interfacing

Jeff Holtzman

■Connecting peripheral devices such as printers, modems, and plotters to personal computers can be confusing. Many books dealing with the subject concentrate on connecting specific computers to specific peripherals.

We cannot guarantee that by absorbing the information presented here you will able to connect any computer to any peripheral device. Designers and manufacturers in the microcomputer world do their best to avoid any efforts at following standards.

What you will learn are the basic principles of both parallel and series interfaces, with hints on how to get two pieces of equipment talking to each other. As Count Basie used to say, let's try it "just one more once."

Serial versus parallel

Most microcomputers move data to and from peripheral devices in eight-bit bytes. There are two methods of moving a byte from one location to another. We can send all eight bits at once, or we can send one bit at a time. The one-bit-at-a-time approach is called serial data transmission, and the all-at-once approach is called parallel data transmission. Neither is better; each has advantages and disadvantages that must be weighed for each application.

If we send one bit at a time, we can get by with as few as two lines (signal and ground) if we only need one way communication, or three lines, if we need communications to and from a peripheral device. We'll talk about that later. But now, consider that for parallel transmission, we're going to need eight separate signal lines, plus a ground, and another one or two for synchronizing things so data won't be lost.

One advantage of parallel transmission is, since all bits are sent at once, transmission may occur at a higher speed than serial transmission. Also, the parallel circuitry is simpler and less expensive than serial circuitry. But the connecting cables are more expensive than serial cables, and there is less standardization of parallel connectors than serial connectors.

In another example, disk drives have (special) parallel interfaces that let them to take advantage of the speed advantage of the parallel approach. Some laboratory equipment and some low cost personal computers communicate with disk drives over (special) serial interfaces, but that is the exception.

Plotters and graphics printers usually operate with parallel interfaces. The time to send—serially—the large amounts of data they require would be excessive.

Parallel transmission

The minimum parallel interface consists of ten signal lines and a ground. Frequently each active (non-ground) line of a parallel interface is twisted together with—or run close by—a ground line. That provides immunity to electrical interference, and is one reason parallel cables often have so many leads.

Of the active lines, eight are for data, one is a STROBE line, which indicates to the receiving device that it should take the data present on the data lines, one is a BUSY signal, which goes high to indicate that the receiving device is busy, and not to transmit any more data until it goes low, and the final signal is called $\overline{\text{ACK}}$, which is (usually) a short, negative-going pulse from the receiving device that indicates the data has been accepted properly. A simple parallel interface might be wired as shown in Fig. 1. Note that the twisted-pair grounds are not shown. Note also that the Busy and $\overline{\text{ACK}}$ lines come from the peripheral device, and that the STROBE line goes to it.

The normal sequence for sending a byte of data over such an interface is as follows. Refer to the timing diagram in Fig. 2. The character in that figure is the letter "U," which has an ASCII value of 55.

The computer monitors the BUSY line, waiting for the peripheral device to OK sending data. After the BUSY lines goes low, the computer places its data on the eight data lines and pulses the STROBE line. That informs the peripheral device to grab the data. After it has taken the data, it pulses the $\overline{\text{ACK}}$ line, and turns the BUSY

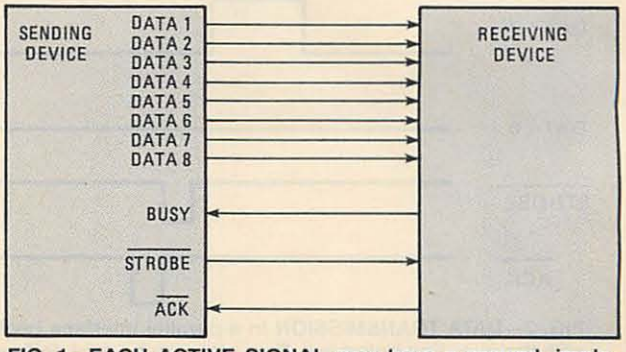


FIG. 1—EACH ACTIVE SIGNAL may have a ground, implemented as a twisted pair, as a shield, or as a nearby wire in a multi-conductor ribbon cable.

line on, if necessary, while processing that byte of data. When it finishes doing that, the BUSY line goes low, and the process may be repeated. If you are interested in how fast a parallel device can accept data, try connecting an oscilloscope to the STROBE line of your computer.

The highest bit, bit 8, is low. ASCII encoded data requires seven bits, so the eighth bit may be ignored or used for other purposes. Some word processing programs use the eighth bit to mark end of words, and in data communications, the eighth bit is often used to indicate the parity of a character.

Parity indicates how many of the lower seven bits of a character are "on" (high). Our "U" has four "on" bits, and four is an even number. If our transmitting device were set up for even parity, bit eight of the "U" would be low, since the character already has four bits. If our transmitter were set up for odd parity, bit eight of the "U" would be high, in order to make a total of 5 "on" bits. A transmitted "V" (ASCII 56, decimal 96) would cause the opposite parity values.

Parity is used for error checking. When a device using parity checking receives a character with incorrect parity, it signals the driving software that an error has occurred, which might then cause a request for the data to be re-transmitted. The most common circuitry allows parity to be set for odd, even, mark, space, or no parity. We have discussed the first two; mark and space simply force bit 8 high or low, respectively; and no parity simply ignores bit 8. Unless you have a specific reason not to do so, choose the latter when setting up your equipment. Otherwise only seven bits of each byte will carry useful data.

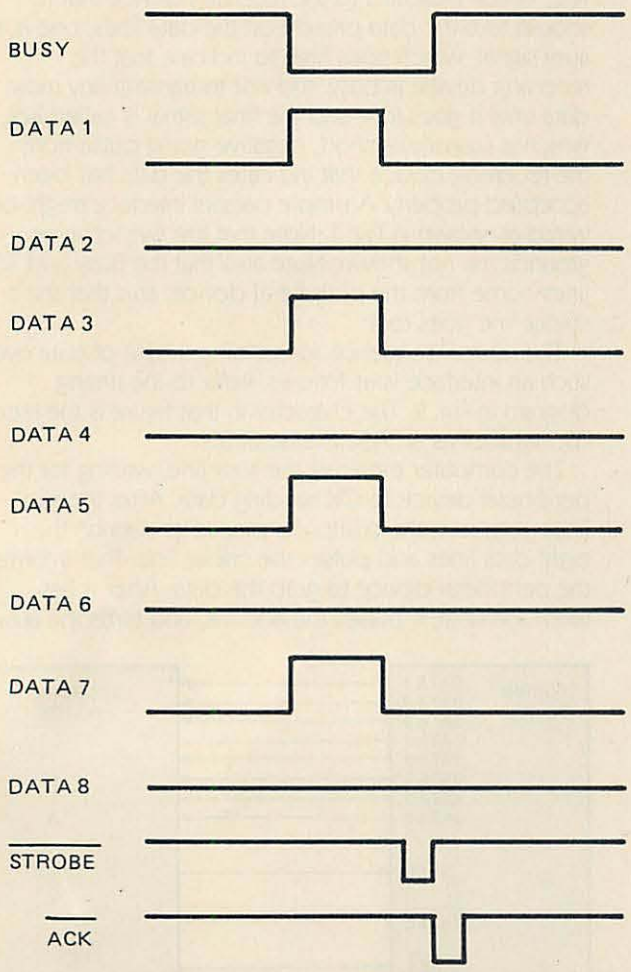


FIG. 2—DATA TRANSMISSION in a parallel interface begins with the BUSY line going low. Data bits may be sent, followed by a STROBE signal. Receipt of the character is acknowledged by sending the $\overline{\text{ACK}}$ line low. The character "U" is shown here.

The transmission process outlined could be applied to any parallel interface: one linking a computer and a printer, a computer and a laboratory instrument, a computer and another computer, etc. However, a typical parallel printer interface will have a number of additional signal lines, as shown in Fig. 3.

The PAPER END signal indicates when the printer is out of paper. The SELECT line is generally high when the printer is "on-line"—that is, ready to receive data, and low when the ON-LINE switch has been pressed by the operator. The $\overline{\text{I-PRIME}}$ signal is a reset line that will force the printer into some well-defined (by each particular printer manufacturer!) state. Finally, the FAULT line goes high when the printer is off-line, when it is out of paper or ribbon, or if some sort of transmission error has occurred.

It is important to understand that not every printer will have all of those signals, and some printers may have additional signals (such as a low-current +5-volt source for powering peripheral devices such as a serial to parallel interface). It is also important to understand that little—if any—microcomputer software makes much use of any of the warning signals shown in Fig. 3. Usually, a printer suffering from a paper-out condition will cause the computer to lock up because the printer sends both the Busy and Paper End signals low, but software usually pays attention only to the Busy line. So your word-processing program might know that the printer is "busy" but not the reason for it.

A serial interface transmits bits of data one after the other. But not at random. When data is transmitted over a line conforming to the RS-232 standard, there are no timing restrictions on when individual bytes of data may be sent. However, each bit of a particular byte must be transmitted with strict attention to timing. Baud Rate refers to the speed at which transmission occurs, but not in a glib way.

Baud rate

First, let's define Baud Rate as the number of bits that may be transmitted per second. A Baud Rate of 300 means that 300 bits may be transmitted per second. That doesn't mean that 300 bits will necessarily be transmitted every second, though that's possible.

The term baud rate refers to the spacing between each bit of a single character, not the speed at which complete characters are transmitted. The way to find the time between each bit is take the inverse of the Baud Rate:

TC (sec) = 1 / Baud Rate

To find the time it takes to transmit a complete character, multiply TC by the number of bits per character.

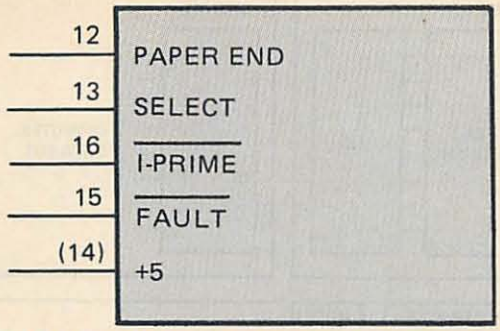


FIG. 3—ADDITIONAL SIGNALS in a parallel interface. These signals are frequently unused by software.

In a 300 baud system, the amount of time from the start of one bit to the start of the next is 1/300, or 3.33 msec. To determine the amount of time a character takes, multiply the number of bits per character by 3.33. How many bits does a character have? In the microcomputer world of the 1980s, the correct answer is usually eight. There are exceptions.

An 8-bit character is going to be 10 or 11 bits long! Every character has a start bit, and one or more stop bits. Most equipment operates with one stop bit; older equipment used two stop bits to give slow machines a little bit (no pun intended) of "recovery" time.

Transmission of a character begins with a start bit. The data bits follow (usually eight, but not always), and then one or two stop bits. Figure 4 shows the transmission of the "U" character with 1 start bit, 1 stop bit and 8 data bits. Note the time "t" between each bit; that is the value 1/Baud rate.

If our 8-bit character has one start and one stop bit, that is a total of 10 bits. So the time to transmit one character at 300 baud is 10 * 3.33 = 33.3 msec.

How many characters can be transmitted in one second, if it takes 33.3 msec to transmit one character? Using a little algebra, we see that

$$\frac{\textbf{1char}}{\textbf{0.0333sec}} = \frac{\textbf{Xchars}}{\textbf{1.0sec}}$$

So X = 1*1/.0333 = 30 chars/sec. We could have arrived at the same answer without the algebra—but doing it the hard way taught us something. Divide the baud rate by the number of bits per character. In our case, and in most you're likely to encounter, just divide the baud rate by ten.

300 / 10 = 30 chars/sec.

What is the character/second rate of a 110-baud transmission?

If you came up with 11, you jumped the gun. You can't really answer the question without knowing the number of bits per character. We picked 110 baud because the old Teletype machines used that as their baud rate. And they used two stop bits—so the TTY has 11 bits/char, and 110/11 = 10 chars/sec.

Here's another trick question: If characters are coming over a transmission line at 300 baud, does that mean that 30 characters come through every second? It doesn't, and that brings us to our next topic.

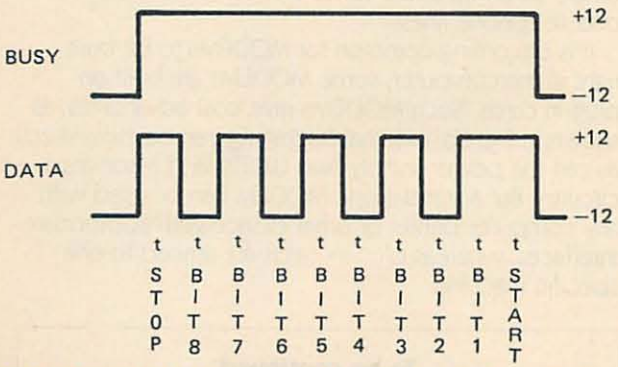


FIG. 4—SERIAL TRANSMISSION of data also begins with the BUSY line going low. A "U" is being transmitted here. Note the time "t" between each bit.

Transmission rate

Transmission rate may be defined as the number of characters flowing over a communications line per unit of time. The most important thing to realize is that there is no relationship between baud rate and transmission rate! You might have one computer dumping data at 300 Baud and another at 9600 Baud, and the latter might have a lower transmission rate than the former. How could that happen?

The 300 baud machine might be dumping 30 characters every second, the 9600 baud machine might only send a character once every 30 minutes. Remember a transmission rate of 300 Baud guarantees only that there will be a delay of 3.33 msec between each bit of a character.

That example illustrates the point. Higher baud rates aren't necessarily better. You could attach circuitry (such as a "printer buffer") to a 110-baud Teletype machine that would enable your computer to dump characters at 9600 baud. The TTY will be unable to print at the rate of 960 characters/second, so as soon as the buffer memory inside your interface filled up, your effective transmission rate would fall from the baud rate to near the actual printing speed of the TTY.

Adding a high-speed interface to a slow piece of equipment may still be an advantage. If the buffer memory could hold a fairly large number of characters, your computer might dump an entire document at high speed and go on to do something else while the buffer outputs data at a rate the printer can handle.

If you attach a buffer memory to your printer (or MODEM, for that matter), make sure you get a buffer with enough memory to make it cost effective. If you mostly print double-spaced documents under about 25 pages, a 64K buffer should suffice. But with documents much longer than that, or with graphics dumps, you'll run up against the same problem. Once the buffer's memory is filled, transmission rate will

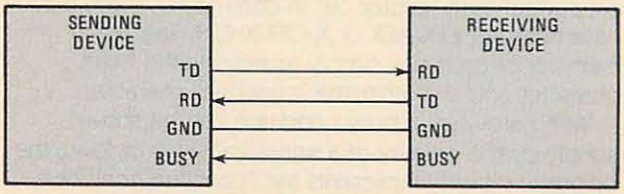


FIG. 5—RS-232 INTERFACES are commonly wired with only the signals shown here.

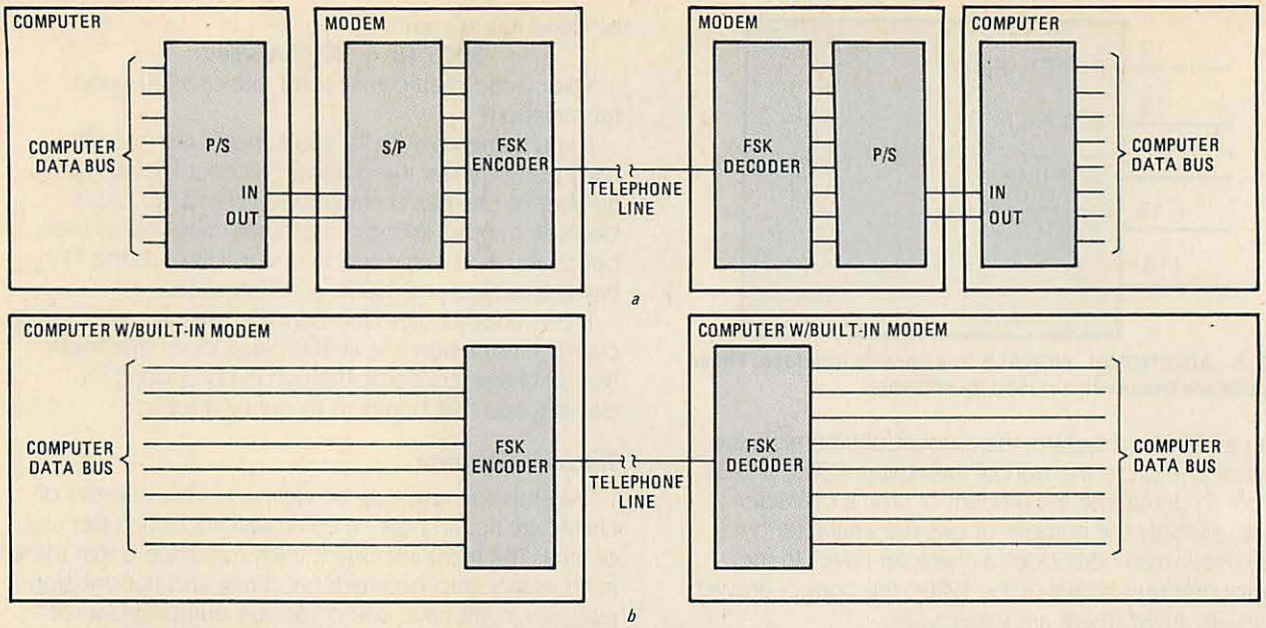


FIG. 6—TWO COMPUTERS connected via external MODEM's over the telephone lines are shown at *a*, and at *b* the computers are linked by internal MODEMs.

decrease drastically.

A practical RS-232 interface would appear as in Fig. 5. There are two data lines, a ground, and, like the parallel interface shown in Fig. 1, a BUSY line. There is no strobe line, as the start bit informs the receiving device that more is on the way. Neither is there an ACK. The BUSY line may perform the same function as in a parallel interface, as shwon in fig. 4, but there are other ways of indicating a busy condition.

Controlling transmission rate

Often we are unable to allow a computer to spew forth data at a transmission rate equal (or close to) the maximum value allowed by the baud rate—30 chars/sec, in the 300-baud system discussed here. The hardware on the receiving end must have a way of telling the transmitter "Hey—wait! I can't accept any more data right now."

There are two ways of doing that: With hardware and with software. With software we require two communication lines: One allowing data to flow from the transmitter to the receiver, and another allowing data to flow from the receiver to the transmitter.

What happens is that the transmitter sends data as fast as it can, while monitoring the line from the receiver for a special "wait" character that tells the transmitter not to send any more data. After receiving the "wait" character, the transmitter stops until it receives a "continue" character from the receiver. There are several such "protocols" in common use; you may have heard of ETX/ACK or X-OFF/X-ON. The first member of each pair simply represents the "wait" character, and the other, the "continue" character.

With hardware, a busy condition can be shown simply by the polarity of a signal line (high or low); the opposite polarity represents the "continue"condition. For example, low represents "Busy," high represents "continue."

Note that neither the hardware nor the software method is better than the other; each was developed to solve different problems. The software protocols for use with MODEMs and remotely operated printers where single-line communication links are a part of the system. A seismographic sensing unit might be located in a remote location, would be linked to a printer in a geological survey office by MODEM. A signal in addition to the transmit data and receive data lines that would distinguish between "wait" and "continue" conditions is hard to implement. Modern printers let you choose between software and hardware solutions, or both.

Historically, MODEMs were serial-in serial-out devices, as shown in Fig. 6a. There we see two computers connected through MODEMs via telephone lines. Each computer has a device labeled "P/S Converter," and each MODEM has a device labeled "S/P Converter." That is the same device, and is called a UART (Universal Asynchronous Receiver Transmitter). Another device inside the MODEM (called an FSK, for Frequency Shift Keying encoder) converts the parallel binary data into audio tones that can be transmitted over telephone lines.

It is becoming common for MODEMs to be built right in the computer; some MODEMs are built on plug-in cards. Such MODEMs give cost advantages, as shown in Fig. 6b: external packaging can be eliminated, as can the power supply, two UARTs, and associated circuitry. But a stand-alone MODEM can be used with any computer, printer or other device with appropriate interfaces, whereas plug-in cards are limited to one specific machine.

To be continued in next month's issue